* 计算机系统层次结构
  + 组成：硬件系统+软件系统
    - 硬件组成：输入设备、输出设备、存储器、运算器、控制器——CPU=运算器+控制器+寄存器
    - 软件分类：系统软件、应用软件
  + 多级层次结构
    - 第1级：微程序机器层，实在的硬件层，由机器硬件直接执行微指令
    - 第2级：传统机器语言层，实际的机器层，由微程序解释机器指令系统
    - 第3级：操作系统层，由操作系统程序实现，也成为混合层
    - 第4级：汇编语言层，由汇编程序支持和执行
    - 第5级：高级语言层，面向用户，由各种高级语言编译程序支持和执行
* 计算机工作过程

1. 将程序和数据装入主存
2. 从程序起始地址运行程序
3. 从程序首地址开始，取指、译码、执行、计算下一条指令地址，直至结束

* 计算机性能指标
  + 机器字长：计算机能进行一次整数运算所能处理的二进制数据的位数
  + 数据通路带宽：外部数据总线一次所能并行传输的位数
  + 主存容量：贮存所能存储信息的最大容量
  + 运算速度：吞吐量、响应时间、CPU时钟周期、主频、CPI、CPU执行时间、MIPS等
* IEEE754标准
  + 短实数：1符号S+8阶码E+23尾数M=32位
  + 长实数：1符号S+11阶码E+52尾数M=64位
* 程序访问局部性原理
  + 时间局部性：如果一个存储项被访问，则该项可能很快会被再次访问
  + 空间局部性：如果一个存储项被访问，则该项及其邻近的项可能很快会被再次访问
* 高级语言到机器语言的翻译方式
  + 编译：运行程序前翻译
    - 预处理：包含头文件、替换宏、删除注释
    - 编译
    - 汇编
    - 链接
  + 解释：运行程序时翻译
* Cache工作基本原理
  + 组成：快表和快速存储器
  + 工作原理：处理器按主存地址访问存储器，存储器地址的高段通过主存-cache地址映射机构借助查块表判定该地址的存储单元是否在cache中，如果在，则命中，按cache地址访问；否则不命中，访问主存，并从著挫女观众调入相应数据块到cache中，若cache中已写满，则按照某种算法将cache中的某一块替换出去，并修改有关的地址映射关系
  + 和主存的映射方式
    - 直接映射：将主存地址映射到cache的一个指定地址
    - 全相联映射：任何主存地址可映射到任何cache地址
    - 组相联映射：将存储空间的页面分成若干组，各组之间直接映射，组内各块全相联映射
* Cache主存块的替换算法

|  |  |  |  |
| --- | --- | --- | --- |
|  | 思想 | 优点 | 缺点 |
| 随机算法RAND | 用软的或硬的随机数产生器产生上层中要被替换的页号 | 简单,易于实现 | 没有利用上层存储器使用的"历史信息",没有反映等程序局部性,命中率低. |
| 先进先出FIFO | 选择最早装入上层的页作为被替换的页 | 实现方便,利用了主存历史的信息 | 不能正确反映程序局部性原理,命中率不高,可能出现一种异常现象. |
| 近期最少使用法LRU | 选择近期最少访问的页作为被替换的页 | 比较正确反映程序局部性,利用访存的历史信息,命中率较高 | 实现较复杂 |
| 优化替换算法OPT | 将未来近期不用的页换出去 | 命中率最高,可作为衡量其他替换算法的标准 | 不现实,只是一种理想算法 |

* 虚拟存储器
  + 定义：具有请求调入和置换功能，能从逻辑上对内存容量加以扩充的一种存储器系统
  + 页式虚拟存储器
    - 页式管理：是把虚拟存储空间和实际空间等分成固定大小的页,各虚拟页可装入主存中的不同实际页面位置.页式存储中,处理机逻辑地址由虚页号和页内地址两部分组成,实际地址也分为页号和页内地址两部分,由地址映射机构将虚页号转换成主存的实际页号
    - 页式管理用一个页表,包括页号,每页在主存中起始位置,装入位等.页表是虚拟页号与物理页号的映射表.页式管理由操作系统进行,对应用程序员的透明的.
  + 段式虚拟存储器
    - 段式管理: 把主存按段分配的存储管理方式.它是一种模块化的存储管理方式,每个用户程序模块可分到一个段,该程序模块只能访问分配给该模块的段所对应的主存空间.段长可以任意设定,并可放大和缩小.
    - 系统中通过一个段表指明各段在主存中的位置.段表中包括段名(段号),段起点,装入位和段长等.段表本身也是一个段.段一般是按程序模块分的
  + 段页式虚拟存储器
    - 段页式管理:是上述两种方法的结合,它将存储空间按逻辑模块分成段,每段又分成若干个页,访存通过一个段表和若干个页表进行.段的长度必须是页长的整数倍,段的起点必须是某一页的起点
  + TLB(快表)
    - 在虚拟存储器中进行地址变换时,需要虚页号变换成主存中实页号的内部地址变换,这一过程
    - 缓存时首先要到主存查页表，然后才能根据主存物理地址访问主存的存取指令或数据。因此采用虚拟存储器机制后，访存的次数增加了。为了减少访存的次数，往往将页表中最活跃的几个页表项复制到高速缓存中。这种在高速缓存中的页表项称为快表（translation look aside buffer）
    - 查表时,根据虚页表同时查找快表和慢表,当在快表中查到该虚页号时,就能很快找到对应的实页号,将其送入主存实地址寄存器,同时使慢表的查找作废,这时主存的访问速度没降低多少.如果在快表中查不到,则经过一个访主存的时间延迟后,将从慢表中查到的实页送入实地址寄存器,同时将此虚页号和对应的实页号送入快表
* 指令的执行过程
  + 读取指令
  + 分析指令
  + 执行指令
  + 检查有无中断请求
* CISC和RISC

|  |  |  |
| --- | --- | --- |
|  | CISC | RISC |
| 指令系统 | 庞杂 | 精简 |
| 指令数 | 一般>200 | 一般<100 |
| 寻址方式 | 不固定 | 固定32位 |
| 可访存指令 | 不加限制 | 只有LOAD/STORE指令 |
| 各种指令使用频率 | 相差很大 | 相差不大 |
| 各种指令执行时间 | 相差很大 | 相差不大 |
| 优化编译实现 | 很难 | 较容易 |
| 程序源码长度 | 较短 | 较长 |
| 控制逻辑实现方式 | 绝大多数为微程序控制 | 绝大多数为硬连线控制，应用流水线技术 |

RISC机优点：

充分利用了VLSI芯片的面积

提高了计算机运行速度

便于设计，降低成本，提高可靠性

有效支持高级语言程序

* 指令流水线基本概念
  + 流水线基本原理：流水线技术已使用显著提高指令执行速度与效率的技术。方法是：指令取值完成后，不等该指令执行完毕即可取下一条指令
  + 影响流水线性能的因素
    - 结构相关：多条指令在同一时刻争用同一资源形成冲突
      * 暂停一个时钟周期
      * 单独设置数据存储器和指令存储器
    - 数据相关：后继指令需要用到前面指令的执行结果
      * 暂停时钟周期
      * 数据旁路：把前一条指令的ALU计算结果直接输入到下一条指令
    - 控制相关：分支指令和改变其他值的指令
* 独立磁盘冗余阵列RAID：把相同的数据存储在多个硬盘的不同的地方（因此，冗余地）的方法。通过把数据放在多个硬盘上，输入输出操作能以平衡的方式交叠，改良性能。因为多个硬盘增加了平均故障间隔时间（MTBF），储存冗余数据也增加了容错
* IO方式
  + 程序查询方式
  + 程序中断方式
  + DMA方式：外设不通过CPU直接与系统内存交换数据